IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received October 11, 2018, accepted November 4, 2018, date of publication November 12, 2018, date of current version December 18, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2880975

A Publicly Verifiable Multi-Secret Sharing Scheme With Outsourcing Secret Reconstruction

CHANGLU LIN[1,2], HUIDAN HU[1,2], CHIN-CHEN CHANG[3], (Fellow, IEEE), AND SHAOHUA TANG[4], (Member, IEEE)

[1]College of Mathematics and Informatics, Fujian Normal University, Fuzhou 350117, China
[2]Fujian Provincial Key Laboratory of Network Security and Cryptology, Fujian Normal University, Fuzhou 350007, China
[3]Department of Information Engineering and Computer Science, Feng Chia University, Taichung 40724, Taiwan
[4]School of Computer Science and Engineering, South China University of Technology, Guangzhou 510006, China

Corresponding author: Changlu Lin (cllin@fjnu.edu.cn)

This work was supported by the National Natural Science Foundation of China under Grants 61572132, U1705264, and 61632013.

ABSTRACT A publicly verifiable secret sharing (PVSS) scheme enjoys the public verification and the lower cost of communication than VSS scheme. However, the existing PVSS schemes cannot be applied in the scenarios of the devices with low computation ability and do not share the multiple secrets among all participants efficiently. In this paper, an efficient publicly verifiable multi-secret sharing scheme with outsourcing secret reconstruction is proposed. Each participant only spends a small amount of computational cost to recover multiple secrets because of the expensive burden of computation and verifiability is outsourced to the cloud service provider (CSP). Moreover, the CSP knows no information of the secrets, and the participants have the abilities to verify the returned result. We also prove that our scheme is secure under the hardness assumption of the discrete logarithm problem and the modified generalized bilinear inversion problem.

INDEX TERMS Data confidentiality, computation integrity, multiple secret sharing, outsourcing computation, verifiable secret sharing.

I. INTRODUCTION

Computation and storage outsourcing to the cloud service provider (CSP) has become a universal phenomenon due to the rapid of development of the cloud computing. The hardware and available memory of a client's devices, such as the cell-phones, the portable laptops, significantly limit the computational capacity and stored ability of processing the numerous data. That a client holding lower computational level is not able to execute the desired computational tasks, he/she would like to have recourse to the cloud service provider to solve the numerous computational tasks or the large-scale storage requirements.

In 1979, Shamir [1] and Blakely [2] proposed the threshold secret sharing scheme to share a secret efficiently and safely based on the Lagrange interpolating polynomial and the linear projective geometry, respectively. However, their schemes exist two drawbacks: (1) the dealer shares only one secret among the participants in every secret sharing process; (2) the dealer and participants are deemed to be honest absolutely. A multi-secret sharing (MSS) scheme was proposed to solve the drawback (1), in which the dealer can share the multiple secrets among the participants in each secret sharing process, and each participant holds one share only. Meanwhile, a kind of secret sharing scheme known as the verifiable secret sharing (VSS) scheme was proposed to repair the drawback (2), in which dishonest behavior of the dealer and the participants can be found in the distribution phase and the reconstruction phase, respectively. Subsequently, the verifiable multi-secret sharing scheme was also presented to make the multi-secret sharing scheme enjoy the property of verifiability.

A publicly verifiable secret sharing (PVSS) scheme has more prominent merits than the verifiable secret sharing (VSS) scheme in some applications. An obvious advantage is the public verification: anyone, who even is the outsider of the scheme, has ability to verify the validity of the shares by using some public messages. Another clear character is the lower cost of communication: the dealer shares the shadows to the participants over an open channel. It is noted that the publicly verifiable multi-secret sharing (PVMSS) scheme is more efficient than the public verifiable secret sharing (PVSS) scheme, because PVMSS scheme has both properties of the public verification and the multiple secret

sharing. Nonetheless, the participants need to undertake the huge amount of computation for recovering the shared secret and verifying the validity of the shares in the secret reconstruction process and the verification process, respectively. Some participants cannot support the cost of computation in both processes, when their devices are low capability of computation, like smart phone, iPad, and so on. If the participants want to recover the secret and check the validity of the shares effectively, then the ability of computation of the participants will be a major concern.

This paper will explore a new publicly verifiable multi-secret sharing scheme and outsource the process of the secret reconstruction to the cloud service provider. Our proposed scheme enjoys four advantages such as *the multiple secret sharing*, *the privacy of the shared secrets*, *the efficient secret reconstruction*, and *the efficient verification of the share and the returned result*. The last two properties are thanking to the outsourcing computation of the secret reconstruction and the share verification. Furthermore, our proposed PVMSS supports the participants to recover the desired return with the help of the cloud service provider even if the participants' computational abilities are limited.

Related works. It is convenient and efficient for the clients to compute and store sensitive data with the help of the cloud service provider. However, the security and privacy of the clients' data become major issues when sensitive data of the clients is performed in an incompletely trusted cloud service provider. In cloud environment, we need to ensure *the data confidentiality*: the cloud service provider cannot get any information of the clients' sensitive data, and*the computation integrity*, which also is called as the result verifiability [3] or the checkability [4]: each client has ability to verify the returned result from the cloud service provider.

Several early results [5]–[7] provide the power to the clients to detect the misbehavior of the cloud service provider; however, the privacy of the clients' confidential data cannot be guaranteed in these schemes. There are several recent works [8]–[10] are classified as secure outsourcing computation because they ensure two significant properties: the result verification and the data privacy. Subsequently, Dong and Ren [11] proposed an efficient and secure outsourcing scheme, in which the cost of verifiable process is reduced. Meanwhile, the works of Fun and Samsudin [12] and Tang *et al.* [13] based on homomorphic encryption to insure the security of outsourcing computation.

Stadler [14] first introduced the notation of publicly verifiable secret sharing (PVSS). This kind of scheme has the special property that anyone including the participants has ability to check the validity of share from the dealer. Very early works [14]–[16] made some contributions in the domain of PVSS. Recently, Jhanwar *et al.* [17] used the technique of Paillier encryption to design a new construction for PVSS. There are several papers based on bilinear maps to improve the efficiency of public verification in [18]–[20] by using the different methods. Short integer solution is used [21] to enhance the scheme's security. Meanwhile, Peng and Tian [22], [23] proposed two simple PVSS schemes based on multilinear Diffie-Hellman assumption to ensure optimal information rate and stronger security. However, at most all PVSS schemes have a common shortage that each participant has to undertake the heavy cost of computation in the verification and reconstruction processes. Zhang *et al.* [24] proposed a new verifiable secret sharing scheme to largely reduce computational cost for the participants, in which the participants only need to undertake few cost of computation by outsourcing computational tasks to the cloud service provider. Recently, Zhang *et al.* [25] designed a cloud storage system for electronic health records (EHRs) from the secret reconstruction outsourcing based on Shamir's secret sharing scheme. A large-scale system of linear equations is outsourced to the cloud service provider, while the verification of health records is realized by a secure hash function.

Our contributions. This paper proposes an efficient PVMSS scheme called as a publicly verifiable multi-secret sharing scheme with outsourcing secret reconstruction. Our scheme guarantees the significant properties: the secret confidentiality and the computation integrity when a number of computational and stored tasks are outsourced to the cloud service provider. Our main contributions are described as follows.

- **Multiple secrets**. Our scheme allows the dealer to share m secrets among the participants, while all participants in authorized set can recover each secret in different stages *without* the predefined order.
- **Secret privacy.** Our scheme ensures the privacy of the shared m secrets. It means that the cloud service provider known as semi-honest model cannot get any information about secrets even if the participants outsource a massive computational tasks to the cloud service provider.
- **High efficiency**. Every participant in our scheme only consume the lower computational cast comparing with recent proposed schemes [17]–[20], [22], [23] (see Table 2) by the method of outsourcing secret reconstruction tasks to the cloud service provider.
- **Result verifiability**. Our scheme provides the verifiable ability for the participants. It implies that each participant can check the returned result from the cloud service provider is true or false.

The paper is organized as follows. In Section II, we introduce the relevant notations including entities, bilinear maps, and so on. Our proposed PVMSS scheme is shown in Section III. Analysis of the proposed PVMSS scheme is given in Section VI. Finally, we conclude the paper in Section V.

II. PRELIMINARIES

In this section, at first, we begin briefly describing the entities including the dealer, the cloud service provider and the participant, and then giving some definitions of the bilinear map, and some hardness assumptions. We also review the model of publicly verifiable secret sharing scheme, and quickly

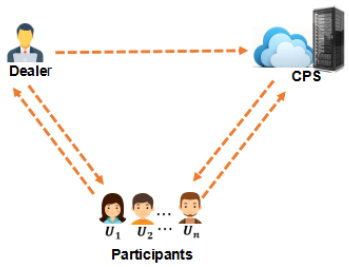

FIGURE 1. The relationship among the dealer, the participants, and the cloud service provider.

introduce the model of the PVMSS with outsourcing secret reconstruction. We finally define the security requirements for the PVMSS scheme.

A. ENTITIES

Entities are classified into the following three types in our proposed PVMSS scheme.

- **Dealer**. The dealer who is not absolutely trusted the third party needs to choose the parameters and to compute the public values and the encrypted pseudo shares. The dealer shares the encrypted pseudo shares to the cloud service provider and send some values to all participants.
- **Cloud service provider (CSP).** Usually, the semi-honest model [26] and/or the stronger security model (such as malicious model) are considered, according to the server's behavior. The cloud service provider is a semi-honest in this paper. It means that the CSP faithfully executes the protocol's steps and thus correctly performs the computational task and returns the result to each client. However, the CSP still tries to know confidential information of the clients. In our paper, the cloud service provider needs to execute the following steps: 1) stores all encrypted pseudo shares and verifies them by the dealer; 2) stores the private keys and verify them by the participants in authorized set; 3) compute sm pseudo secrets by using pseudo shares and returns them to the participants.
- **Participant**. There are n participants in our scheme, denoted as the set $U = \{U_1, U_2, \ldots, U_n\}$. Each participant needs to randomly choose two private keys (d_i, t_i) and compute two public keys (P_i, T_i) shared to the dealer in initialization phase for $i = 1, 2, \ldots, n$. Meanwhile, the participants compute a few message for sharing the corresponding private key d_i to the cloud service provider in secret reconstruction phase.

B. BILINEAR MAPS

We denote G_1 and G_2 as an additive cyclic group and a multiplicative cyclic group, respectively. Let G_1 and G_2 have the same order q, where q a large prime. A map $e : G_1 \times G_1 \rightarrow G_2$ is known as a bilinear map if it satisfies the three conditions as follows.

- **Bilinearity:** for all $P_1, P_2 \in G_1$ and $a_1, a_2 \in Z_q^*$, we have $e(a_1P_1, a_2P_2) = e(P_1, P_2)^{a_1a_2}$.
- **Non-degeneracy:** there exist $P_1, P_2 \in G_1$ such that $e(P_1, P_2) \neq 1$.
- **Computability:** for all $P_1, P_2 \in G_1$, there exists an efficient algorithm to compute $e(P_1, P_2)$.

C. HARDNESS ASSUMPTIONS

Some well-known hardness assumptions will be considered when we analyze the security of our scheme. Here we list three hardness assumptions as follows.

- **Computational Diffie-Hellman (CDH) problem:** Given $P, aP, bP \in G_1$, it is hard to generate $abP \in G_1$ for some $a, b \in Z_q^*$.
- **Discrete logarithm (DL) problem:** Given $P, aP \in G_1$, it is difficult to compute $a \in Z_q^*$.
- **Modified generalized bilinear inversion (MGBI) problem:** Given $\varphi \in G_2$, and a generator P of G_1, it is hard to find $\mathrm{Q} \in G_1$ such that $e\,(\mathrm{Q},P) = \varphi$.

D. THE MODEL

In this subsection, we describe the model of publicly verifiable secret sharing (PVSS) scheme which consists of the initialization phase, the distribution phase and the reconstruction phase described as follows. We also compare the different aspects of our proposed PVMSS scheme with outsourcing secret reconstruction with the traditional PVSS scheme.

- **Initialization phase**: the dealer executes the setup algorithm **Stp**. It outputs the public and common values Vs by inputting a security parameter 1^λ and the set of participants $U = \{U_1, U_2, \ldots, U_n\}$. It is written as $\mathrm{Vs} \leftarrow \mathbf{Stp}(1^\lambda, P)$. Each participant U_i generates his public key pk_i according to his private key sk_i and some public values.
- **Distribution phase:** the secret is divided by dealer in this phase and the dealer distributes the shares by a special way. Each share can be verifiable publicly.
  1) **Secret distribution:** the dealer performs the distribution algorithm **Dist**. It outputs the encrypted share $\{Y_i\}_{i\in P}$ and the public parameters Pms used to verify the encrypted shares by inputting some public values and the secret S. It is written as $(\{Y_i\}_{i\in P}, Pms) \leftarrow \mathbf{Dist}(\mathrm{Vs}, S)$.
  2) **Public verifiability of the share:** everyone, including all participants, carries out the verification algorithm **Ver** by checking whether or not the equation $\mathbf{Ver}(\mathbf{Vs}, Pms, \{Y_i\}_{i\in P}) = \text{true}$.
- **Reconstruction phase:** All the decrypted shares are sent to the corresponding participants and are checked the integrity before they are accepted by each participant in authorized set. If all checks are correct, then the secret is reconstructed correctly.
  1) **Decryption of the encrypted share:** every participant U_i in authorized set uses the

corresponding private key to decrypt the encrypted share Y_i for the receiving share s_i.

2) **Verification of the share:** every participant P_i in authorized set verifies the shares s_j from other participants in authorized set.
3) **Reconstruction of the secret:** the participants in authorized set carry out the reconstruction algorithm **Rec** to recover the secret S. It is written as $S \leftarrow \mathbf{Rec}(\{s_j\}_{i\in A})$, where A is the authorized set.

Our proposed PVMSS scheme with outsourcing secret reconstruction is also constituted by above three phases. However, there exist some differences comparing with the traditional PVSS scheme. Mainly different aspects are described as follows: 1) the dealer shares m pseudo secrets $S'_h(h = 1, 2, \cdots, m)$ among the participants in one time, and the encrypted pseudo shares $c'_i(i = 1, 2, \cdots, n)$ are sent to the cloud service provider by the dealer; 2) the cloud service provider assists the participants in authorized set to recover m pseudo secrets $S'_h(h = 1, 2, \cdots, m)$ (not the true secrets), and then sends them to each participant in authorized set. But, the cloud service provider knows nothing information about m secrets $S_h(h = 1, 2, \cdots, m)$ since the cloud service provider does not have the ability to get the significant message S''_h, where $S_h = A_h \oplus S'_h \oplus S''_h$.

E. SECURITY REQUIREMENTS

A PVMSS with outsourcing secret reconstruction scheme must satisfy the security requirements described as follows.

- **Correctness**: if the dealer and the participants act honestly, each participant in authorized set can recover the same secret by pooling all shares from all authorized participants.
- **Verifiability:** the dishonest behaviors of the dealer or participants can be checked by any verifier after the distribution phase and before the reconstruction phase.
- **Privacy:** a) it is impossible for an honest dealer, the participant of any unauthorized set to get any information of the secret; b) unrecovered secrets cannot be know from the recovered secrets; c) the cloud service provider known as semi-honest model cannot learn any information about the secrets.
- **Result verifiability:** each participants enable to checking whether the returned result from the cloud service provider is true or not.

III. THE PROPOSED PVMSS SCHEME

The basic idea of our proposed scheme is that the true secrets are hidden in the public values $A_h = S_h \oplus S'_h \oplus S''_h$ for $h = 1, 2, \cdots, m$ where S'_h are the pseudo secrets, and S''_h only knew by the participant, while the secret privacy and result verifiability are ensured under the help of the cloud service provider who only has ability to recover the pseudo secrets S'_h. Furthermore, each participant verifies the returned result S'^*_h from the CSP by checking the equation $h\left(S'^*_h\right) = h(S'_h)$ to guarantee the integrity of result. Each participant also recovers the true secrets S_h by computing $S_h = A_h \oplus S'_h \oplus S''_h$.

A. INITIALIZATION PHASE

In this subsection, we assume that there are n participants denoted as $\mathrm{U} = \{U_1, U_2, \cdots, U_n\}$ and there are m seccrets will be shared among n participants. The dealer and participants execute the followings steps.

1) The dealer generates a bilinear maps $e{:}G_1 \times G_1 \rightarrow G_2$ among G_1 and G_2 and the generator P of the group G_1, where G_1 is additive cyclic group and G_2 is multiplicative cyclic group and a secure hash function $h(\cdot)$.
2) The dealer chooses two polynomials with $t-1$ degree $f(x) = \sum_{j=1}^{t-1} a_j x^j$ and $g(x) = \sum_{j=1}^{t-1} b_j x^j$ for $a_j, b_j \in Z_q^*$.
3) The dealer chooses the public parameters $n_h, r_h \in Z_q^*$ for $h = 1, 2, \cdots, m$ and secret parameter $r \in Z_q^*$ randomly, and then computes the public values $A_h = S_h \oplus S'_h \oplus S''_h$ for $h = 1, 2, \cdots, m$ where $S'_h = e(P, P)^{a_0 + r r_h b_0}$, $S''_h = e(P, P)^{n_h}$, $a_0 = f(0)$ and $b_0 = g(0)$.
4) The participant U_i chooses two private keys $t_i, d_i \in Z_q^*$ randomly, and then computes the public keys $P_i = d_i P$ and $T_i = t_i P$ for $i = 1, 2, \cdots, n$.

B. DISTRIBUTION PHASE

In this stage, the dealer first generates encrypted pseudo shares and sends them to cloud service provider by an open channel. The dealer then computes the parameters for the participants. The steps are as follows.

(a) The encrypted pseudo share distribution

1) The dealer generates the encrypted pseudo shares $c'_i = e((f(i) + r r_h g(i)) P_i, P)$ and the parameters $V_i = e(n_h T_i, P)$ for $i = 1, 2, \cdots, n$.
2) The dealer computes public values $C_j = a_j P$, $D_j = r_h b_j P^*$ and $h(S'_h)$ for $j = 1, 2, \cdots, t-1$ where $P^* = rP$. These public values will be used to check whether the encrypted pseudo shares and pseudo secrets are true or not.
3) The dealer sends all encrypted pseudo shares c'_i to cloud service provider and the parameter V_i to each participant U_i by an open channel.

(b) Public verifiability of the encrypted pseudo share

1) After receiving all encrypted pseudo shares, the cloud service provider needs to check the validity of encrypted pseudo shares. The cloud service provider runs verification algorithm to check whether the following equation is held or not.

$$c'_i = e(X_i, P_i) \cdot e(Y_i, P_i) \quad \text{for } i = 1, 2, \cdots, n.$$

Here we get $X_i = f(i) \cdot P$ and $Y_i = r_h g(i) P^*$ by using public values $C_j = a_j P$ and $D_j = r_h b_j P^*$, as follows:

$$X_i = f(i) \cdot P = \sum_{j=0}^{t-1} a_j \cdot (i)^j \cdot P = \sum_{j=0}^{t-1} (i)^j \cdot C_j$$

and $Y_i = r_h \cdot g(i) \cdot P^* = \sum_{j=0}^{t-1} r_h \cdot (i)^j \cdot b_j \cdot P^* = \sum_{j=0}^{t-1} (i)^j \cdot D_j$.

2) After receiving the parameter V_i, the participant U_i gets the parameters S_h'' by computing $S_h'' = (V_i)^{t_i^{-1}}$. This is because of $(V_i)^{t_i^{-1}} = \mathrm{e}(n_h T_i, P)^{t_i^{-1}} = \mathrm{e}(n_h t_i P, P)^{t_i^{-1}} = \mathrm{e}(n_h P, P)^{t_i^{-1} t_i} = S_h''$.

C. OUTSOURCING SECRET RECONSTRUCTION

In this process, assume t participants U_i, for $i = 1, 2, \cdots, t$, in some authorized set want to recover m secrets. There are two stages: a) the cloud service provider needs to decrypt the encrypted pseudo shares after receiving the private keys from the participants in authorized set; b) all authorized participants recover m secrets after receiving pseudo secrets from CSP with very low computation. The steps are shown as follow.

(a) Decryption of the encrypted pseudo share

1) The cloud service provider computes the messages $m_i = e(z_i P_i, P)$ for $z_i \in Z_q^*$ and sends them to t participants $U_i (i = 1, 2, \cdots, t)$.
2) After receiving the message m_i, each participant U_i computes m_i' and m_i'', where $m_i' = (m_i)^{d_i^{-1}} = e(z_i P, P)$ and $m_i'' = d_i \cdot e(z_i P, P)$ for $i = 1, 2, \cdots, t$.
3) After receiving the message sm_i'', the cloud service provider gets private keys d_i from the private values $e(z_i P, P)^{-1}$, where $d_i = m_i'' \cdot e(z_i P, P)^{-1}$.
4) After recovering the private keys d_i, the cloud service provider needs to check the private keys d_i by computing L_i, where $L_i = z_i \cdot d_i \cdot P$. The cloud service provider then checks whether the equation $\mathrm{e}(L_i, P) = \mathrm{e}(P_i, z_i P)$ is held or not.
5) The cloud service provider uses the private keys d_i to compute the pseudo shares s_i' as follows:

$$\begin{aligned} s_i' &= (c_i')^{d_i^{-1}} \\ &= \mathrm{e}((f(i) + r r_h g(i)) P_i, P)^{d_i^{-1}} \\ &= \mathrm{e}((f(i) + r r_h g(i) d_i P, P)^{d_i^{-1}} \\ &= \mathrm{e}((f(i) + r r_h g(i) P, P)^{d_i^{-1} d_i} \\ &= \mathrm{e}((\mathrm{f}(i) + r r_h g(i) P, P). \end{aligned}$$

(b) Reconstruction of the secrets

1) The cloud service provider uses t shares s_i', for $i = 1, 2, \cdots, t$, to recover m pseudo secrets S_h' by using Lagrange interpolation as following equation:

$$S_h' = \prod_{i=1}^{t} (s_i')^{\beta_i} = e(P, P)^{a_0 + r r_h b_o},$$

where $\beta_i = \prod_{j \neq i} \frac{i}{j - i}$ is Lagrange coefficient.

2) After receiving the messages $S_h'^*(h = 1, 2, \cdots, m)$ from the cloud service provider, each participant first checks whether $h\left(S_h'^*\right) = h(S_h')$. If $h\left(S_h'^*\right) = h(S_h')$ is held, then $S_h'^* = S_h'$. It implies that the computational result of the cloud service provider is correct; otherwise the result is wrong. Then, every participant computes the secrets $S_h = A_h \oplus S_h'^* \oplus S_h''$ for $h = 1, 2, \cdots, m$. Each participant U_i in authorized set only is required to spend a small amount of computational task to recover all valid secrets.

IV. ANALYSIS OF OUR PVMSS SCHEME

In this section, we first give the security analysis of our proposed PVMSS scheme by confirming Lemma 1 to Lemma 5 and Theorem 1 to Theorem 3. We also compare our proposed PVMSS scheme with other existing PVSS schemes in the properties and computational cost of each participant.

A. SECURITY ANALYSIS

Lemma 1: Each participant and the cloud service provider verify the validity of encrypted pseudo shares c_i' by the equation: $c_i' = e(X_i, P_i) \cdot e(Y_i, P_i)$.

Proof: We use $C_j = a_j P$ and $D_j = r_h b_j P^*$ to compute X_i and Y_i respectively.

$$\begin{aligned} X_i &= f(i) \cdot P = \sum_{j=0}^{t-1} a_j \cdot (i)^j \cdot P \\ &= \sum_{j=0}^{t-1} (i)^j \cdot C_j, \text{ and} \\ Y_i &= r_h \cdot g(i) \cdot P^* = \sum_{j=0}^{t-1} r_h \cdot (i)^j \cdot b_j \cdot P^* \\ &= \sum_{j=0}^{t-1} (i)^j \cdot D_j. \end{aligned}$$

We then gain the following value,

$$\begin{aligned} c_i' &= e((f(i) + r r_h g(i)) P_i, P) \\ &= e((f(i) + r r_h g(i)) d_i P, P) \\ &= e(P, P)^{(f(i) + r r_h g(i)) d_i} \\ &= e(P, P)^{f(i) d_i} \cdot e(P, P)^{r r_h g(i) d_i} \\ &= \mathrm{e}(f(i) d_i P, P) \cdot e(r r_h g(i) d_i P, P) \\ &= \mathrm{e}(f(i) P, d_i P) \cdot \mathrm{e}(r r_h g(i) d_i P, P) \\ &= \mathrm{e}(f(i) P, d_i P) \cdot \mathrm{e}(r r_h g(i) P^*, d_i P) \\ &= e(X_i, P_i) \cdot e(Y_i, P_i). \end{aligned}$$

Hence, if the equation holds, then the cloud service provider conforms that all encrypted pseudo shares from the dealer are valid.

Lemma 2: The cloud service provider checks the validity of the private keys d_i from the participants by the equation $\mathrm{e}(L_i, P) = \mathrm{e}(P_i, z_i P)$ for $i = 1, 2, \cdots, n$.

Proof: After computing the value L_i, where $L_i = z_i d_i P$, we compute

$$\mathrm{e}(L_i, P) = e(z_i d_i P, P) = e(d_i P, z_i P) = e(P_i, z_i P).$$

Hence, if the equation holds, then the cloud service provider confirms that the private key from the corresponding participant are valid.

Remark 1: Lemmas 1 and 2 show that the cloud service provider getting the messages s_i' is true by the method of using the private key d_i to decrypt the encrypted pseudo shares c_i'.

Lemma 3: All participants in authorized set recover m pseudo secrets correctly.

Proof: In other words, it is need to verify equation $S'_h = \prod_{i=1}^{t} (s'_i)^{\beta_i} = e(P,P)^{a_0+rr_hb_o}$, where $\beta_i = \prod_{j\neq i} \frac{i}{j-i}$. We have the following equation,

$$\begin{aligned}\prod_{i=1}^{t} (s'_i)^{\beta_i} &= \prod_{i=1}^{t} e((f(i) + rr_h g(i))P, P)^{\beta_i} \\ &= e(P,P)^{\sum_{i=0}^{t-1}(f(i)+rr_h g(i))\cdot\beta_i} \\ &= e(P,P)^{\sum_{i=0}^{t-1} f(i)\cdot\beta_i} \cdot e(P,P)^{\sum_{i=0}^{t-1} rr_h g(i)\cdot\beta_i} \\ &= e(P,P)^{f(0)} \cdot e(P,P)^{rr_h g(0)} \\ &= e(P,P)^{f(0)+rr_h g(0)} \\ &= e(P,P)^{a_0+rr_h b_o}.\end{aligned}$$

It is easy to check the equation is held. It implies that the recovered pseudo secrets are correct.

Lemma 4: The cloud service provider known as semi-honest model learns no information about the secrets $S_h (h = 1, 2, \cdots, m)$.

Proof: The cloud service provider has ability to recover m pseudo secrets S'_h by computing $S'_h = \prod_{i=1}^{t} (s'_i)^{\beta_i} = e(P,P)^{a_0+rr_hb_o}$. However, the secrets $S_h = A_h \oplus S'_h \oplus S''_h$, the parameters S''_h cannot be known by the cloud service provider. So the cloud service provider is not enable to recover m secrets by pseudo secrets S'_h and the public values A_h.

Lemma 5: Each participant in the authorized set checks the returned result S'^{*}_h from the cloud service provider by the equation $h\left(S'^{*}_h\right) = h(S'_h)$.

Proof: After receiving the messages S'^{*}_h, the participants check whether the equation $h\left(S'^{*}_h\right) = h(S'_h)$. If the cloud service provider can find the value S'^{*}_h $(S'^{*}_h \neq S'_h)$ to make $h\left(S'^{*}_h\right) = h(S'_h)$, it is contradiction with the hash function which is collision resistant.

Theorem 1: In the proposed PVMSS scheme with outsourcing secret reconstruction, an adversary cannot know any information about the private keys d_i, the pseudo shares s'_i and the secrets S_h by using the public values C_j, D_j, P_i, A_h and $h(S'_h)$ under DL hardness assumption.

Proof: 1) Security of the private keys d_i and pseudo shares s'_i. The adversary cannot get the private keys d_i from the public keys P_i. If the adversary gets d_i from P_i, he is able to solve DL problem in G_1, which is contradiction with the hardness assumption of DL. Obviously, the adversary has not ability to know the pseudo shares s'_i even if he has the encrypted pseudo shares c'_i from the public values C_j and D_j. Because an adversary cannot learn the private keys d_i to decrypt encrypted pseudo shares c'_i.

2) Security of Pseudo Secrets. We first find that the adversary cannot know the pseudo secrets S'_h from the public values $h(S'_h)$. If the adversary can get S'_h by using the values $h(S'_h)$, which is contradiction with secure one-way function $h(\cdot)$. Meanwhile, the public values A_h are useless for the adversary when he has nothing information about S'_h.

Remark 2. Theorem 1 confirms that all public values do not weaken the security of our PVMSS scheme which is computational security under DL hardness assumption.

Theorem 2: A probabilistic polynomial time adversary cannot recover the unrecovered pseudo secrets by using the reveled shares and the pseudo recovered secrets under MGBI hardness assumption in the proposed PVMSS scheme.

Proof: Assume that $k (k < m)$ secrets $S'_1, S'_2, \cdots, S'_k$ have been recovered. Consequently, the values $S'_j = e((a_0 + r_j rb_0)P, P)$ and $s'_{i_j} = e((f(i) + r_j rg(i))P, P)$ are shown for $i = 1, 2, \cdots, t; j = 1, 2, \cdots, k$. Unrevealed $m - k$ secrets may be got by the adversary, if he has ability to know the messages $a_0 + r_j rb_0$ or $f(i) + r_j rg(i)$ from the values $S'_j = e((a_0 + r_j rb_0)P, P)$ and $s'_{i_j} = e((f(i) + r_j rg(i))P, P)$, respectively.

Case 1: An adversary gets a_0 and rb_0 by solving two linear equations $a_0 + r_j rb_0$ and $a_0 + r_z rb_0$ where $j \neq z$ for $j, z \in \{1, 2, \cdots, k\}$. An adversary can compute unrevealed $m - k$ secrets $S'_{k+1}, S'_{k+2}, \cdots, S'_m$ by computing the values $e((a_0 + r_{k+1}rb_0)P, P), \cdots, e((a_0 + r_m rb_0)P, P)$. However, an adversary cannot know the message $a_0 + r_j rb_0$ from $S'_j = e((a_0 + r_j rb_0)P, P)$. If not, it is contradiction with MGBI problem.

Case 2: An adversary gets $f(i)$ and $rg(i)$ by two solving linear equations $f(i) + r_j rg(i)$ and $f(i) + r_z rg(i) j \neq z$ for $j, z \in \{1, 2, \cdots, k\}$. It means that an adversary can compute t pseudo shares $s'_{1_h} = e((f(1) + r_h rg(1))P, P), s'_{2_h} = e((f(2) + r_h rg(2))P, P), \cdots, s'_{t_h} = e((f(t) + r_h rg(t))P, P)$ for $h = k + 1, k + 2, \cdots, m$ corresponding to $m - k$ pseudo secrets $S'_{k+1}, S'_{k+2}, \cdots, S'_m$. The reason is same as Case 1, an adversary cannot know the message $f(i) + r_j rg(i)$ from $s'_{i_j} = e((f(i) + r_j rg(i))P, P)$.

In a word, we confirm that unrevealed pseudo secrets cannot be gained by an adversary using the recovered pseudo secrets $S'_j = e((a_0 + r_j rb_0)P, P)$ and the revealed shares $s'_{i_j} = e((f(i) + r_j rg(i))P, P)$ in our scheme.

Theorem 3: Any $t - 1$ participants cannot recover the pseudo secrets S'_j for $j = 1, 2, \cdots, m$ in our PVMSS scheme.

Proof: Assume that $t-1$ participants $U_1, U_2, \cdots, U_{t-1}$ want to recover secret S'_j. For computing $S'_j = e((a_0 + r_j rb_0)P, P)$ they require knowledge of at least t values of the shares $s'_{1_j} = e((f(i) + r_j rg(i))P, P), s'_{2_j} = e((f(i) + r_j rg(i))P, P), \cdots, s'_{t_j} = e((f(i) + r_j rg(i))P, P)$ for $j = 1, 2, \cdots, m$. Since for any $s'^{*}_{t_j} = e((f'(i) + r_j r'g'(i))P, P)$ and $t - 1$ the share $ss'_{i_j} = e((f(i) + r_j rg(i))P, P)i = 1, 2, \cdots, t - 1$, there is a unique polynomial $Q'_j(x)P$ of degree $t - 1$ where $Q'_j(x)P = (f'(x) + r_j r'g'(x))$ such that $s'_{1_j} = e(Q'_j(i)P, P), s'_{2_j} = e(Q'_j(2)P, P), \cdots, s'_{t-1_j} = e(Q'_j(t - 1)P, P)$ and $s'_{t_j} = e(Q'_j(t)P, P)$. It means that any $t - 1$ shares $s'_{i_j} = e((f(i) + r_j rg(i))P, P)$ cannot ensure a unique value

TABLE 1. The property comparison with previous PVSS schemes.

Schemes	Multiple secrets	Outsourcing
in [17]	×	×
in [18]	×	×
in [19]	√	×
in [20]	×	×
in [22]	×	×
in [23]	×	×
This work	√	√

$Q_j(0)P$. Consequently, $t-1$ participants cannot compute the secrets S'_j in our proposed scheme.

B. PERFORMANCES AND COMPARISONS

In this subsection, we mainly present performance result about the number of secrets among the participants and the computational cost of each participant by comparing our scheme with some recent related works. The results are shown in Table 1 and Table 2. For the convenience of evaluating the computational cost, we define some notations described as follows:

TG_e : the time of executing a bilinear pairing operation e:$G_1 \times G_1 \rightarrow G_2$.

TG_e^m: the time of executing a multilinear pairing operation e:$G_1^m \rightarrow G_2$.

TG_{mul}: the time of executing a scalar multiplication operation of point in G_1.

TG_{mul}^m: the time of executing a scalar multiplication operation of point in G_2.

T_{exp}: the time of executing a modular exponentiation operation.

Table 1 shows that our proposed publicly verifiable multi-secret sharing scheme has an important merit of multi-secrets. It means that the scheme only is executed one time to share m secrets among the participants, while each participant just needs to keep two private keys for m secrets. However, when the works in [17], [18], [20], [22], and [23] share m secrets among the participants, each participant has to keep m shares for recovering m different secrets and the dealer needs to repeatedly execute the scheme for sharing m secrets. It indicates that the burden of the dealer and participants will be increased by the method of [17], [18], [20], [22], and [23].

Meanwhile, we show another significant merit in our scheme by comparing with [17]–[22] in Table 2. We notice that in the share verification process, each participant must consume the computational cost of $(t+3)\,T_{exp}$, $tTG_{mul}+2TG_e$, $4TG_e+(t+1)\,TG_{mul}+T_{exp}$, $lT_{exp}+2TG_e$, $tT_{exp}+2TG_e$ and $(t+1)\,TG_{mul}+2TG_e^m$ respectively in [17]–[22]. However, the participants in our scheme does not need to verify the share since this work is done by CSP and the computational cost is zero. Moreover, the participants in the scheme only execute a modular exponentiation operation

TABLE 2. The computational cost comparison with previous PVSS schemes.

Schemes	Verifiability phase	Reconstruction phase
in [17]	$(t+3)\,T_{exp}$	tT_{exp}
in [18]	$tTG_{mul}+2TG_e$	$2TG_e+tTG_{mul}$
in [19]	$4TG_e+(t+1)TG_{mul}+T_{exp}$	$(t+1)T_{exp}$
in [20]	$lT_{exp}+2TG_e$	$(t+1)T_{exp}$
in [22]	$tT_{exp}+2TG_e$	$(t+1)TG_{mul}^m$
in [23]	$(t+1)TG_{mul}+2TG_e^m$	$(t+1)T_{exp}$
This work	0	$1T_{exp}$

to recover the shared secret, while each participant has to spend a large number of computational cost of tT_{exp}, $2TG_e+tTG_{mul}$, $(t+1)\,T_{exp}$, $(t+1)\,T_{exp}$, $(t+1)\,TG_{mul}^m$ and $(t+1)\,T_{exp}$ respectively in [17]–[22]. Obviously, our scheme largely reduces the computational cost of participants with the help of cloud service provider. Each participant consumes a small amount of computation to check the share and recover the secret. While, in [17]–[22], every participant has greater burden of computation than the participant in our scheme.

V. CONCLUSIONS

In this paper, we presented a novel publicly verifiable multi-secret sharing scheme and outsourced the massive computational tasks to the cloud service provider. In the proposed scheme, we use the cloud service provider to support the expensive tasks of reconstruction computation and verifiable computation. Thus the participants can efficiently recover the multiple secrets with a small amount of computational cost. Meanwhile, we use the messages A_h and S''_h to hide the true secret S_h, where $S_h = A_h \oplus S'_h \oplus S''_h$ for insuring the cloud service provider knows no information about the true secret. Each authorized participant can verify the returned result by the public hash values. Furthermore, our proposed scheme archives several significant merits: the multiple secret sharing, the privacy of the shared secrets, the efficient secret reconstruction, and the efficient verification of the share and the returned result.

REFERENCES

[1] A. Shamir, "How to share a secret," *Commun. ACM*, vol. 22, no. 11, pp. 612–613, 1979.
[2] G. R. Blakley, "Safeguarding cryptographic keys," *Proc. Nat. Comput. Conf.*, vol. 48, pp. 313–317, Feb. 1979.
[3] R. Gennaro, C. Gentry, and B. Parno, "Non-interactive verifiable computing: Outsourcing computation to untrusted workers," in *Advances in Cryptology* (Lecture Notes in Computer Science). Berlin, Germany: Springer, 2010.
[4] S. Hohenberger and A. Lysyanskaya, "How to securely outsource cryptographic computations," in *Theory of Cryptography Conference* (Lecture Notes in Computer Science). Berlin, Germany: Springer, 2005.
[5] W. Du, M. Murugesan, and J. Jia, "Uncheatable grid computing," in *Algorithms and Theory of Computation Handbook*. London, U.K.: Chapman & Hall, 2010, p. 30.

[6] S. Goldwasser, Y. T. Kalai, and G. N. Rothblum, "Delegating computation: Interactive proofs for muggles," in *Proc. 14th Annu. ACM Symp. Theory Comput.*, 2008, pp. 113–122.
[7] P. Golle and I. Mironov, "Uncheatable distributed computations," in *Topics in Cryptology* (Lecture Notes in Computer Science). Berlin, Germany: Springer, 2001, pp. 425–440.
[8] X. F. Chen, J. Li, J. Ma, Q. Tang, and W. Lou, "New algorithms for secure outsourcing of modular exponentiations," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 9, pp. 2386–2396, Sep. 2014.
[9] G. Xu, G. T. Amariucai, and Y. Guan, "Delegation of computation with verification outsourcing: Curious verifiers," *IEEE Trans. Parallel Distrib. Syst.*, vol. 28, no. 3, pp. 717–730, Mar. 2017.
[10] Z. Chen, A. Fu, K. Xiao, M. Su, Y. Yu, and Y. Wang, "Secure and verifiable outsourcing of large-scale matrix inversion without precondition in cloud computing," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2018, pp. 1–6.
[11] M. Dong and Y. Ren, "Efficient and secure outsourcing of bilinear pairings with single server," *Sci. China Inf. Sci.*, vol. 61, no. 3, p. 039104, 2018.
[12] T. S. Fun and A. Samsudin, "A survey of homomorphic encryption for outsourced big data computation," *KSII Trans. Internet Inf. Syst.*, vol. 10, no. 8, pp. 3826–3851, 2016.
[13] J. Tang, Y. Cui, Q. Li, K. Ren, J. Liu, and R. Buyya, "Ensuring security and privacy preservation for cloud data services," *ACM Comput. Surv.*, vol. 49, no. 1, 2016, Art. no. 13.
[14] M. Stadler, "Publicly verifiable secret sharing," in *Advances in Cryptology* (Lecture Notes in Computer Science). Berlin, Germany: Springer, 1996, pp. 190–199.
[15] E. Fujisaki and T. Okamoto, "A practical and provably secure scheme for publicly verifiable secret sharing and its applications," in *Advances in Cryptology* (Lecture Notes in Computer Science). Berlin, Germany: Springer, 1998, pp. 32–46.
[16] B. Schoenmakers, "A Simple publicly verifiable secret sharing scheme and its application to electronic voting," in *Advances in Cryptology* (Lecture Notes in Computer Science). Berlin, Germany: Springer, 1999, pp. 148–164.
[17] M. P. Jhanwar, A. Venkateswarlu, and R. Safavi-Naini, "Paillier-based publicly verifiable (non-interactive) secret sharing," *Des., Codes Cryptogr.*, vol. 73, no. 2, pp. 529–546, 2014.
[18] Y. Tian, C. Peng, and J. Ma, "Publicly verifiable secret sharing schemes using bilinear pairings," *Int. J. Netw. Secur.*, vol. 14, no. 3, pp. 142–148, 2012.
[19] T.-Y. Wu and Y.-M. Tseng, "Publicly verifiable multi-secret sharing scheme from bilinear pairings," *IET Inf. Secur.*, vol. 7, no. 3, pp. 239–246, Sep. 2013.
[20] S. Mashhadi, "Secure publicly verifiable and proactive secret sharing schemes with general access structure," *Inf. Sci.*, vol. 378, pp. 99–108, Feb. 2017.
[21] M. H. Dehkordi and R. Ghasemi, "A lightweight public verifiable multi secret sharing scheme using short integer solution," *Wireless Pers. Commun.*, vol. 91, no. 3, pp. 1459–1469, 2016.
[22] Q. Peng and Y. Tian, "Publicly verifiable secret sharing scheme and its application with almost optimal information rate," *Secur. Commun. Netw.*, vol. 9, no. 18, pp. 6227–6238, 2016.
[23] Q. Peng and Y. Tian, "A publicly verifiable secret sharing scheme based on multilinear diffie-hellman assumption," *Int. J. Netw. Secur.*, vol. 18, no. 6, pp. 1192–1200, 2016.
[24] E. Zhang, J. Peng, and M. Li, "Outsourcing secret sharing scheme based on homomorphism encryption," *IET Inf. Secur.*, vol. 12, no. 1, pp. 94–99, 2018.
[25] H. Zhang, J. Yu, C. Tian, P. Zhao, G. Xu, and J. Lin, "Cloud storage for electronic health records based on secret sharing with verifiable reconstruction outsourcing," *IEEE Access*, vol. 6, pp. 40713–40722, 2018.
[26] O. Goldreich, S. Micali, and A. Wigderson, "How to play any mental game," in *Proc. 19th Symp. Theory Comput. (STOC)*, May 1987, pp. 218–229

CHANGLU LIN received the B.S. and M.S. degrees in mathematics from Fujian Normal University, China, in 2002 and 2005, respectively, and the Ph.D. degree in information security from the State Key Laboratory of Information Security, Graduate University of Chinese Academy of Sciences, China, in 2010. He is currently with the College of Mathematics and Informatics, Fujian Normal University, where he is also with the Fujian Provincial Key Laboratory of Network Security and Cryptology. He is interested in cryptography and network security. He has conducted research in diverse areas, including secret sharing, multi-party computation, public key cryptography, and their applications.

HUIDAN HU received the B.S. degree in mathematics from Yangtze Normal University, China, in 2016. She is currently pursuing the master's degree in applied mathematics with Fujian normal university. Her current research interests include secret sharing and its applications.

CHIN-CHEN CHANG (M'88–SM'92–F'99) received the B.S. degree in applied mathematics and the M.S. degree in computer and decision sciences from National Tsing Hua University and the Ph.D. degree in computer engineering from National Chiao Tung University. From 1989 to 2005, he was with National Chung Cheng University. Since 2005, he has been the Chair Professor with the Department of Information Engineering and Computer Science, Feng Chia University. His research interests include database design, computer cryptography, image compression, and data structures. He is a fellow of the IEE, U.K.

SHAOHUA TANG received the B.S. and M.S. degrees in applied mathematics and the Ph.D. degree in communication and information system from the South China University of Technology in 1991, 1994, and 1998, respectively. Since 2004, he has been a Full Professor with the School of Computer Science and Engineering, South China University of Technology. His current research interests include information security, networking, and information processing.